A NOTE ON THE GENERALIZED MIN-SUM SET COVER PROBLEM

MARTIN SKUTELLA AND DAVID P. WILLIAMSON

ABSTRACT. In this paper, we consider the generalized min-sum set cover problem, introduced by Azar, Gamzu, and Yin [1]. Bansal, Gupta, and Krishnaswamy [2] give a 485-approximation algorithm for the problem. We are able to alter their algorithm and analysis to obtain a 28-approximation algorithm, improving the performance guarantee by an order of magnitude. We use concepts from α-point scheduling to obtain our improvements.

1. INTRODUCTION

In this note, we consider the generalized min-sum set cover problem. In this problem we are given as input a universe U of n elements, a collection $\mathcal{S} = \{S_1, \ldots, S_m\}$ of subsets S_i of U, and a covering requirement $K(S)$ for each $S \in \mathcal{S}$, where $K(S) \in \{1, 2, \ldots, |S|\}$. The output of any algorithm for the problem is an ordering of the n elements. Let C_S be the position of the $K(S)$th element of S in the ordering. The goal is to find an ordering that minimizes $\sum_{S \in \mathcal{S}} C_S$. This problem is a generalization of the min-sum set cover problem (in which $K(S) = 1$ for all $S \in \mathcal{S}$), introduced by Feige, Lovász, and Tetali [3], and the min-latency set cover problem (in which $K(S) = |S|$ for all $S \in \mathcal{S}$), introduced by Hassin and Levin [4]. This generalization was introduced by Azar, Gamzu, and Yin [1] in the context of a ranking problem.

Because the problem is NP-hard, Azar, Gamzu, and Yin give an $O(\log r)$-approximation algorithm for the problem, where $r = \max_{S \in \mathcal{S}} |S|$. This was improved to a constant factor randomized approximation algorithm by Bansal, Gupta, and Krishnaswamy [2]. They introduce a new linear programming relaxation for the problem and show how to use randomized rounding to achieve a performance guarantee of 485.[1] In this paper, we show that by altering their algorithm using some concepts from α-point scheduling (see Skutella [6] for a survey), we are able to improve their algorithm and obtain a performance guarantee of about 28, which is an order of magnitude better.[2]

We now briefly review their algorithm and analysis, and then state the ideas we introduce to obtain an improvement. Their algorithm begins with solving the following linear programming relaxation of the problem, where the variable $y_{S,t}$ for $t \in [n]$ (here and in the following the set $\{1, \ldots, n\}$ is denoted by $[n]$) and set $S \in \mathcal{S}$ indicates whether $C_S < t$ or not, and $x_{e,t}$ for $e \in U$ and $t \in [n]$ indicates whether element e is assigned to the tth

Date: July 12, 2011.

[1] They observe in their paper that they did not try to optimize the constants in their analysis.

[2] Here we would like to point out that $28 \in O(\sqrt{485})$.

position of the ordering:

$$\begin{aligned}
\min \quad & \sum_{t\in[n]} \sum_{S\in\mathcal{S}} (1 - y_{S,t}) \\
\text{s.t.} \quad & \sum_{e\in U} x_{e,t} = 1, && \text{for all } t \in [n], \\
& \sum_{t\in[n]} x_{e,t} = 1, && \text{for all } e \in U, \\
& \sum_{e\in S\setminus A} \sum_{t'<t} x_{e,t'} \geq (K(S) - |A|) \cdot y_{S,t}, && \text{for all } S \in \mathcal{S},\ A \subseteq S,\ t \in [n], \\
& x_{e,t}, y_{S,t} \in [0,1], && \text{for all } e \in U,\ S \in \mathcal{S},\ t \in [n].
\end{aligned}$$

Bansal et al. observe that the exponentially many constraints can be separated in polynomial time such that the linear program can be solved efficiently. Let x^*, y^* be an optimal solution. The algorithm proceeds in a sequence of $\lceil \log n \rceil$ stages. In the ith stage, the algorithm of Bansal et al. computes a probability $p_{e,i} := \min\{1, 8 \sum_{t<2^i} x^*_{e,t}\}$ for each element $e \in U$ by taking the amount that element e is fractionally scheduled up to time 2^i and boosting it by a factor of 8. With probability $p_{e,i}$ it includes element e in a set O_i. If $|O_i| > 16 \cdot 2^i$, the algorithm randomly chooses $16 \cdot 2^i$ elements from O_i and discards the remainder from O_i. For each i, the algorithm picks an arbitrary order for the elements in O_i, then schedules the elements in the order $O_1, O_2, \ldots, O_{\lceil \log n \rceil}$. Notice that it is possible that an element will appear in more than one O_i and is scheduled more than once; one can compute an ordering that keeps only the first occurrence of each element.

The analysis of Bansal et al. works by looking at a time t^*_S for each $S \in \mathcal{S}$, which is the smallest t such that $y^*_{S,t} > 1/2$. The analysis then shows that for any stage i with $t^*_S \leq 2^i$, with probability at least $1 - e^{-1}$ at least $K(S)$ elements have been marked in stage i and no elements are discarded from O_i. From this, the analysis infers that $E[C_S] \leq 64 \cdot \frac{e}{e-2} \cdot t^*_S$. Since the value of the linear program is at least $\frac{1}{2} \sum_{S\in\mathcal{S}} t^*_S$, the paper derives that the expected value of the solution is at most $128 \cdot \frac{e}{e-2} \approx 484.4$ times the value of the linear program.

While we still use several ideas from their algorithm and analysis, we modify it in several key ways. In particular, we discard the idea of stages, and we use the idea of a random α-point for each element e; in particular, after modifying the solution x^* to a solution x in a way similar to theirs, we then randomly choose a value $\alpha_e \in [0,1]$ for each $e \in U$. Let t_{e,α_e} be the first time t for which $\sum_{t'=1}^{t} x_{e,t'} \geq \alpha_e$. We then schedule elements e in the order of nondecreasing t_{e,α_e}. The improvements in analysis come from scrapping the stages (so we don't need to account for the possibility of O_i being too large) and using α-point scheduling; in particular, we introduce a parameter α and look for the last point in time $t_{S,\alpha}$ in which $y^*_{S,t} < \alpha$ (the Bansal et al. paper uses $\alpha = 1/2$). Choosing α randomly gives our ultimate result. We turn to the full analysis in the next section.

2. The Algorithm and Analysis

Let x^*, y^* be an optimum solution to the linear program above. Let $Q > 0$ be a constant determined later. Construct a new solution x from x^* as follows: Initialize $x := Q \cdot x^*$; for $t = 1$ to $\lfloor n/2 \rfloor$ set

$$x_{e,2t} := x_{e,2t} + x_{e,t} \ .$$

Lemma 1. *For each $t \in [n]$*

$$\sum_{t'=1}^{t} \sum_{e\in U} x_{e,t'} \leq 2 \cdot Q \cdot t \ . \tag{1}$$

Moreover, for each $e \in U$ and $t \leq \lfloor n/2 \rfloor$

$$\sum_{t'=t+1}^{2t} x_{e,t'} \geq Q \sum_{t'=1}^{t} x^*_{e,t'} \ , \tag{2}$$

and for each $t \in [n]$

$$\sum_{t'=1}^{t} x_{e,t'} \geq Q \sum_{t'=1}^{t} x^*_{e,t'} \ . \tag{3}$$

Proof. We start by giving an alternative view on the definition of x above. Notice that

$$x_{e,t'} = Q \sum_{t'' \in I(t')} x^*_{e,t''} \qquad \text{with } I(t') := \{t'' : t' = 2^i \cdot t'' \text{ for some } i \geq 0\}. \tag{4}$$

That is, $I(t')$ is precisely the subset of indices t'' such that $x^*_{e,t''}$ contributes to $x_{e,t'}$. For a fixed $t \in [n]$ and $t'' \leq t$, let $J(t'')$ be the subset of all indices $t' \leq t$ such that $x^*_{e,t''}$ contributes to $x_{e,t'}$, i. e., $J(t'') = \{t' \leq t : t'' \in I(t')\}$. By definition of $I(t')$ and $J(t'')$ we get $\sum_{t'=1}^{t} |I(t')| = \sum_{t''=1}^{t} |J(t'')|$. Also notice that $|J(t'')| = 1 + \lfloor \log(t/t'') \rfloor$. Thus,

$$\begin{aligned}
\frac{1}{Q} \sum_{t'=1}^{t} \sum_{e \in U} x_{e,t'} &= \sum_{t'=1}^{t} \sum_{t'' \in I(t')} \underbrace{\sum_{e \in U} x^*_{e,t''}}_{=1} = \sum_{t'=1}^{t} |I(t')| = \sum_{t''=1}^{t} |J(t'')| \\
&= t + \sum_{t''=1}^{t} \lfloor \log(t/t'') \rfloor \leq t + \int_0^t \lfloor \log(t/\theta) \rfloor \, d\theta \\
&= t + \sum_{i=0}^{\infty} \int_{t/2^{i+1}}^{t/2^i} \lfloor \log(t/\theta) \rfloor \, d\theta = t + \sum_{i=0}^{\infty} \frac{t}{2^{i+1}} \cdot i = 2t \ .
\end{aligned}$$

This concludes the proof of (1).

In order to prove (2), simply notice that for each $t'' \in \{1, \ldots, t\}$ there is $t' \in \{t+1, \ldots, 2t\}$ such that $t'' \in I(t')$; then (2) follows from (4). Finally, (3) also follows from (4) since $t' \in I(t')$ for all t'. □

Algorithm: As discussed above, for each $e \in U$ we independently choose $\alpha_e \in [0,1]$ randomly and uniformly. Let t_{e,α_e} denote the first point in time t when $\sum_{t'=1}^{t} x_{e,t'} \geq \alpha_e$. In our final solution, we sequence the elements $e \in U$ in order of nondecreasing t_{e,α_e}; ties are broken arbitrarily.

For $S \in \mathcal{S}$ and some fixed $\alpha \in (0,1)$, let $t_{S,\alpha}$ be the last point in time t for which $y^*_{S,t} < \alpha$. We observe that the contribution of set S to the objective function of the linear program is

$$C^{LP}_S := \sum_{t \in [n]} (1 - y^*_{S,t}) = \int_0^1 t_{S,\alpha} \, d\alpha \ , \tag{5}$$

since in time step t it holds that $t_{S,\alpha} \geq t$ for all $\alpha \in [0,1]$ such that $\alpha > y^*_{S,t}$, or for $(1 - y^*_{S,t})$ of the possible α.

We now bound the probability that we have fewer than $K(S)$ elements from S with $t_{e,\alpha_e} \leq t_{S,\alpha}$ in terms of Q and α.

Lemma 2. *Suppose $Q \cdot \alpha \geq 1$. Define p such that*

$$p := \exp\left(-\frac{1}{2} \cdot \left(1 - \frac{1}{Q \cdot \alpha}\right)^2 \cdot Q \cdot \alpha \right) \leq 1 \ .$$

Then for integer $i \geq 0$,

$$\Pr\left[\left| \{e \in S : t_{e,\alpha_e} \leq 2^i \cdot t_{S,\alpha}\} \right| < K(S) \right] \leq p^{i+1} \ .$$

Proof. Our analysis follows some of the analysis of Bansal et al. for a stage. Let

$$A := \left\{ e \in S : \sum_{t' \le 2^i \cdot t_{S,\alpha}} x_{e,t'} \ge 1 \right\}.$$

Then observe that for any $e \in A$ it holds that $\Pr[t_{e,\alpha_e} \le 2^i \cdot t_{S,\alpha}] = 1$. By the properties of the linear program,

$$\sum_{e \in S \setminus A} \sum_{t' \le t_{S,\alpha}} x^*_{e,t'} \ge \big(K(S) - |A|\big) \cdot y^*_{S,1+t_{S,\alpha}} \ge \big(K(S) - |A|\big) \cdot \alpha \,,$$

so that by (3)

$$\sum_{e \in S \setminus A} \sum_{t' \le t_{S,\alpha}} x_{e,t'} \ge \big(K(S) - |A|\big) \cdot Q \cdot \alpha \,.$$

More generally, it follows from induction on i and (3) and (2), that

$$\sum_{e \in S \setminus A} \sum_{t' \le 2^i \cdot t_{S,\alpha}} x_{e,t'} \ge (i+1) \cdot \big(K(S) - |A|\big) \cdot Q \cdot \alpha \,.$$

For any $e \in S \setminus A$, let random variable X_e be 1 if $t_{e,\alpha_e} \le 2^i \cdot t_{S,\alpha}$ and 0 otherwise. Note that $\Pr[X_e = 1] = \sum_{t' \le 2^i \cdot t_{S,\alpha}} x_{e,t'}$. Let $X := \sum_{e \in S \setminus A} X_e$ and $\mu := E[X]$, so that

$$\mu = E[X] = \sum_{e \in S \setminus A} \sum_{t' \le 2^i \cdot t_{S,\alpha}} x_{e,t'} \ge (i+1) \cdot \big(K(S) - |A|\big) \cdot Q \cdot \alpha \,.$$

Note that if $|A| \ge K(S)$, then $\Pr\left[|\{e \in S : t_{e,\alpha_e} \le 2^i \cdot t_{S,\alpha}\}| < K(S)\right] = 0$, so we assume that $|A| < K(S)$. Then

$$\begin{aligned}
&\Pr\left[|\{e \in S : t_{e,\alpha_e} \le 2^i \cdot t_{S,\alpha}\}| < K(S)\right] \\
&\quad = \Pr\left[|\{e \in S \setminus A : t_{e,\alpha_e} \le 2^i \cdot t_{S,\alpha}\}| < K(S) - |A|\right] \\
&\quad = \Pr\left[X < K(S) - |A|\right] \\
&\quad \le \Pr\left[X < \frac{\mu}{(i+1) \cdot Q \cdot \alpha}\right] = \Pr\left[X < \mu \cdot \left(1 - \left(1 - \frac{1}{(i+1) \cdot Q \cdot \alpha}\right)\right)\right] \\
&\quad \le \exp\left(-\frac{1}{2} \cdot \left(1 - \frac{1}{(i+1) \cdot Q \cdot \alpha}\right)^2 \cdot \mu\right) \\
&\quad \le \exp\left(-\frac{1}{2} \cdot \left(1 - \frac{1}{(i+1) \cdot Q \cdot \alpha}\right)^2 \cdot (i+1) \cdot Q \cdot \alpha\right) \\
&\quad \le \exp\left(-\frac{1}{2} \cdot \left(1 - \frac{1}{Q \cdot \alpha}\right)^2 \cdot (i+1) \cdot Q \cdot \alpha\right) = p^{i+1}
\end{aligned}$$

where we use the Chernoff bound $\Pr[X < \mu \cdot (1-\beta)] \le \exp(-\frac{1}{2} \cdot \beta^2 \cdot \mu)$ (see, for example, Motwani and Raghavan [5, Section 4.1]), and the fact that

$$-\left(1 - \frac{1}{(i+1) \cdot Q \cdot \alpha}\right)^2 \le -\left(1 - \frac{1}{Q \cdot \alpha}\right)^2$$

for $i \ge 0$ and $Q \cdot \alpha \ge 1$. □

Let C_S be a random variable giving the position of the $K(S)$th element of S in the ordering we construct, and let C_S^{LP} be the contribution of set S to the objective function as defined in (5). Then we can bound the cost of our schedule as follows, where $OPT_{LP} = \sum_{S \in \mathcal{S}} C_S^{LP}$ and OPT is the cost of an optimal schedule.

Lemma 3. *If Q and α are chosen such that $p < 1/2$, then*

$$E\left[\sum_S C_S\right] \leq \frac{2 \cdot Q}{1-\alpha} \cdot \frac{1-p}{1-2p} \cdot OPT_{LP} + OPT\ .$$

Proof. Let t_S be the first point in time when $|\{e \in S : t_{e,\alpha_e} \leq t_S\}| \geq K(S)$. Then by Lemma 2, we know that the probability that $t_{S,\alpha} < t_S \leq 2 \cdot t_{S,\alpha}$ is at most p, since the probability that $t_S > t_{S,\alpha}$ is at most p by itself. Similarly, the probability that $2 \cdot t_{S,\alpha} < t_S \leq 4 \cdot t_{S,\alpha}$ is at most p^2, the probability that $4 \cdot t_{S,\alpha} < t_S \leq 8 \cdot t_{S,\alpha}$ is at most p^3, and so on, so that

$$\mathrm{E}[t_S] \leq t_{S,\alpha} + t_{S,\alpha} \sum_{i=0}^{\infty} 2^i \cdot p^{i+1} = t_{S,\alpha} \cdot \left(1 + \frac{p}{1-2p}\right) = t_{S,\alpha} \cdot \frac{1-p}{1-2p}\ . \tag{6}$$

Note that for all $t \leq t_{S,\alpha}$ it holds that $1 - y^*_{S,t} > 1 - \alpha$, so that $C^{LP}_S \geq t_{S,\alpha}(1-\alpha)$, or $t_{S,\alpha} \leq C^{LP}_S/(1-\alpha)$. Thus

$$\mathrm{E}[t_S] \leq C^{LP}_S \cdot \frac{1}{1-\alpha} \cdot \frac{1-p}{1-2p}\ .$$

Observe that $C_S \leq |\{e \in U \setminus S : t_{e,\alpha_e} \leq t_S\}| + K(S)$. Note that for any fixed element $e \notin S$ and time t, the probability that $t_{e,\alpha_e} \leq t$ is $\min\{1, \sum_{t' \leq t} x_{e,t'}\}$, so that

$$E\left[|\{e \in U \setminus S : t_{e,\alpha_e} \leq t\}|\right] \leq \sum_{e \in U} \sum_{t' \leq t} x_{e,t'} \leq 2Q \cdot t$$

by (1). Then we have that

$$\mathrm{E}[C_S] \leq 2Q \cdot \mathrm{E}[t_S] + K(S) \leq \frac{2Q}{1-\alpha} \cdot \frac{1-p}{1-2p} \cdot C^{LP}_S + K(S)\ , \tag{7}$$

from which it follows that

$$\mathrm{E}\left[\sum_S C_S\right] \leq \frac{2Q}{1-\alpha} \cdot \frac{1-p}{1-2p} \cdot OPT_{LP} + OPT\ ,$$

since in any solution $\sum_{S \in \mathcal{S}} K(S) \leq OPT$. □

We try to tune the various parameters to obtain the best possible performance guarantee. If we set $\alpha := 1/2$ (as did Bansal et al. [2]) and $Q := 10.05$, then $p = 0.1995$, and thus we obtain

$$\frac{2Q}{1-\alpha} \cdot \frac{1-p}{1-2p} + 1 \approx 54.54\ ,$$

for a performance guarantee of about 55. However, we can do better if we choose α (and Q) randomly.

Theorem 1. *If we choose α independently at random from $(0,1)$ according to the density function $f(\alpha) = 2 \cdot \alpha$ and set $Q := z/\alpha$ for some appropriately chosen constant z, then the algorithm has performance guarantee less than 27.78.*

Proof. Notice that $\alpha \cdot Q$ is equal to the fixed constant z and $p = \exp\left(-\frac{1}{2} \cdot \left(1 - \frac{1}{z}\right)^2 \cdot z\right)$ depends only on z and is thus constant.

In the proof of Lemma 3 we have obtained bounds on the expectations of t_S and C_S under the assumption that the values of α and Q are fixed. We refer to these conditional expectations by E_α such that

$$\mathrm{E}_\alpha[t_S] \leq t_{S,\alpha} \cdot \frac{1-p}{1-2p} \qquad \text{due to (6), and}$$

$$\mathrm{E}_\alpha[C_S] \leq 2Q \cdot \mathrm{E}_\alpha[t_S] + K(S) \qquad \text{due to (7).}$$

Unconditioning together with (5) then yields

$$\begin{aligned}
\mathrm{E}[C_S] &= \int_0^1 f(\alpha) \cdot E_\alpha[C_S]\, d\alpha \\
&\le \int_0^1 2\alpha \cdot 2Q \cdot t_{S,\alpha} \cdot \frac{1-p}{1-2p}\, d\alpha + K(S) \\
&= 4z \cdot \frac{1-p}{1-2p} \int_0^1 t_{S,\alpha}\, d\alpha + K(S) \\
&= 4z \cdot \frac{1-p}{1-2p} \cdot C_S^{LP} + K(S)\ .
\end{aligned}$$

Thus, we get

$$\mathrm{E}\left[\sum_{S\in\mathcal{S}} C_S\right] \le 4z \cdot \frac{1-p}{1-2p} \cdot OPT_{LP} + OPT \le \left(1 + 4z \cdot \frac{1-p}{1-2p}\right) \cdot OPT\ .$$

If we set $z := 5.03$, then $p \approx 0.1990$, and the performance guarantee is less than 27.78. □

Acknowledgements. The first author was supported by the DFG Research Center MATHEON "Mathematics for key technologies" in Berlin. This work was carried out while the second author was on sabbatical at TU Berlin. He wishes to acknowledge that he was supported in part by the Berlin Mathematical School, the Alexander von Humboldt Foundation, and NSF grant CCF-0830519.

REFERENCES

[1] Y. Azar, I. Gamzu, and X. Yin. Multiple intents re-ranking. In *Proceedings of the 41st Annual ACM Symposium on the Theory of Computing*, pages 669–678, 2009.

[2] N. Bansal, A. Gupta, and R. Krishnaswamy. A constant factor approximation algorithm for generalized min-sum set cover. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1539–1545, 2010.

[3] U. Feige, L. Lovász, and P. Tetali. Approximating min sum set cover. *Algorithmica*, 40:219–234, 2004.

[4] R. Hassin and A. Levin. An approximation algorithm for the minimum latency set cover problem. In G. S. Brodal and S. Leonardi, editors, *Algorithms - ESA 2005, 13th Annual European Symposium, Palma de Mallorca, Spain, October 3-6, 2005, Proceedings*, volume 3669 of *Lecture Notes in Computer Science*. Springer, Berlin, Germany, 2005.

[5] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[6] M. Skutella. List scheduling in order of α-points on a single machine. In *Efficient Approximation and Online Algorithms: Recent Progress on Classical Combinatorial Optimization Problems and New Applications*, volume 3484 of *Lecture Notes in Computer Science*, pages 250–291. Springer, Berlin, Germany, 2006.

INSTITUT FÜR MATHEMATIK, SEKR. MA 5-2, TECHNISCHE UNIVERSITÄT BERLIN, STRASSE DES 17. JUNI 136, 10623 BERLIN, GERMANY.

E-mail address: martin.skutella@tu-berlin.de

SCHOOL OF OPERATIONS RESEARCH AND INFORMATION ENGINEERING, CORNELL UNIVERSITY, ITHACA, NY 14853, USA.

E-mail address: dpw@cs.cornell.edu